Research Paper
Computational Methods for Differential Equations
http://cmde.tabrizu.ac.ir
Vol. *, No. *, *, pp. 1-14
DOI:10.22034/cmde.2026.73558.3762

A Random-Forest-Guided coupled selection of decomposition order and Padé degree for Padé-resummed adomian series: The Fisher–KPP traveling wave

Sathish Marakonda* **and Y. Rajashekhar Reddy**

Department of Mathematics, Jawaharlal Nehru Technological University Hyderabad, Telangana, India.

Abstract

A truncated Adomian series for the Fisher–KPP traveling wave stops improving after a handful of terms; replacing the truncated sum with a rational Padé approximant is a known remedy, but the two quantities that govern its accuracy — how many series terms to generate and what Padé degree to fit through them — are ordinarily chosen by hand for each new problem. We formulate this choice as a constrained search: find the cheapest Padé degree pair (L, N) whose approximant meets a prescribed accuracy tolerance, then train a random forest regressor to predict that pair directly from three problem parameters (reaction rate, wave-front location, and evaluation time), so that the search need not be repeated for every new configuration. The underlying series is generated symbolically from a rescaled autonomous recursion, evaluated on 45 sampled configurations, and used to train the regressor with a 75/25 holdout split. Across the sampled range, the Padé route meets a 10^{-3} tolerance on 42 of 45 configurations against 34 of 45 for size-matched plain truncation, a direct, countable improvement over doing without resummation at all. Four configurations withheld entirely from training were then tested: the regressor reproduced the cost-optimal degree pair exactly in three instances, twice rescuing a case where plain truncation missed tolerance and once finding a genuinely cheaper configuration where truncation already succeeded, and in the fourth instance rounded to a configuration one cost unit short of optimal, which itself missed tolerance — a result reported as obtained, without adjustment, since a coarse rounding error of exactly this kind is informative about where the method's limits lie. Held-out prediction of the combined cost was strong ($R^2 = 0.934$); the numerator degree alone was harder to predict in isolation ($R^2 = 0.384$), consistent with its narrow admissible range on the sampled grid.

Uncorrected

Keywords. Adomian decomposition, Padé approximation, Fisher–KPP equation, Random forest regression, Traveling wave, Parameter selection.
2010 Mathematics Subject Classification. 65L05, 34K28, 35K57, 41A21, 68T05.

1. Introduction

Series solutions produced by the Adomian decomposition method [3, 4] do not always behave well far from their point of expansion. Convergence of the method has been studied under conditions relating successive component norms [1, 8], conditions that are sufficient but not necessary; a well-documented instance of where they are not met in practice is the traveling-wave solution of the Fisher–KPP equation: successive Adomian components stop shrinking after only a handful of terms, so no amount of additional truncation recovers accuracy, a stall confirmed directly in this paper by the classical-truncation columns of Tables 4–7 below, where plain truncation up to $M = 9$ fails outright on two of the four withheld test configurations. Stability of ADM-based numerical schemes more broadly has itself been the subject of dedicated study [14], and manual regularization strategies for problematic series have been proposed for specific problem classes [22], without addressing the coupled decomposition-order/resummation-degree question posed here. ADM's applicability across nonlinear differential equations generally is well established [15, 23], as are alternative semi-analytical constructions such as He's polynomials [11] and the related homotopy perturbation technique [13], though none of these alternatives specifically targets the traveling-wave divergence addressed here.

Received: 14 July 2026; Accepted: 29 October 2026.
∗ Corresponding author. Email: sathishmarakonda@gmail.com.

Rational resummation of the truncated series — replacing $\sum_{n=0}^{M} u_n$ with a Padé approximant fitted through the same coefficients — is a standard way to push past this kind of stall, and has been used successfully for Fisher's equation specifically [21], building on a broader literature applying ADM to systems of linear and nonlinear PDEs [20] in which Padé resummation is noted as one of several established extensions for accelerating or extending series convergence [10]. What is common across this literature is that both the number of series terms generated and the Padé degree fitted through them are supplied by the author, typically after some manual experimentation, rather than determined by a stated criterion.

Manual selection of ADM's governing parameters is the norm across this literature: decomposition order is fixed by the author's own trial and error [22], and where resummation is used at all, its degree is chosen the same way. This paper treats that choice as a constrained search rather than a manual one, and does so specifically for the case — traveling-wave solutions — where fixing the decomposition order alone cannot help, because the underlying series does not converge at any order.

This paper takes that step. The approach followed here is deliberately narrower than embedding a learned model directly into the solution process, as in physics-informed neural networks trained to satisfy a governing PDE directly [19]: the ADM-Padé technique itself is left completely unchanged, and only the two integers governing it — the numerator degree L and denominator degree N of the approximant — are selected jointly against an accuracy tolerance rather than fixed by hand. A search for prior work combining machine-learned parameter selection with Padé-resummed ADM in any setting returned nothing; the coupled, tolerance-verified selection problem posed here appears not to have been formulated before.

The remainder of the paper is organized as follows. Section 2 states the general ADM-Padé coupled selection problem, then specializes it to the Fisher–KPP traveling wave, derives the corresponding Adomian recursion in a form amenable to fast symbolic computation, and describes the training data, drawing on standard numerical-analysis background [7] and, for the fractional-order extension discussed in Section 6, the standard fractional-calculus reference [17]. Section 3 reports the trained regressor's holdout performance, using the random forest architecture standard in the statistical learning literature [5, 6, 12]. Section 4 applies the trained model to four configurations withheld from training. Section 5 discusses what the results do and do not establish, and Section 6 concludes.

2. Formulation

2.1. General formulation: ADM and the coupled Padé-degree selection problem.

The method developed in this paper is not specific to any one equation, and is stated here in general form before being applied, in the remainder of Section 2, to a single case study. For a nonlinear PDE written in Adomian's operator form

$$L(u) + R(u) + N(u) = g(x,t), \tag{2.1}$$

with L the highest-order linear operator, R the remaining linear part, and N the nonlinear part, applying the inverse operator L^{-1} gives

$$u = \phi + L^{-1}(g) - L^{-1}(Ru) - L^{-1}(Nu), \tag{2.2}$$

with ϕ fixed by the initial or boundary conditions. The Adomian decomposition method represents the solution as $u = \sum_{n=0}^{\infty} u_n$ and the nonlinear term as $N(u) = \sum_{n=0}^{\infty} A_n$, where the A_n are the Adomian polynomials generated systematically from the structure of N and depend only on $u_0, \ldots, u_n$. Substituting into (2.2) gives the standard recursion

$$u_0 = \phi + L^{-1}(g), \qquad u_{n+1} = -L^{-1}(Ru_n) - L^{-1}(A_n), \quad n \geq 0, \tag{2.3}$$

and the truncated approximation $u^{(M)} = \sum_{n=0}^{M} u_n$.

For problems where $u^{(M)}(x,t)$ can be organized as a power series in t with coefficients $c_n(x)$ — as holds for any autonomous problem of the form (2.1) with no explicit t-dependence in g, a property verified for the specific case studied here in Section 2.3 — the truncated series can be resummed as an $[L/N]$ Padé approximant,

$$P_{L,N}(x,t) = \frac{\sum_{i=0}^{L} a_i(x)\, t^i}{1 + \sum_{j=1}^{N} b_j(x)\, t^j}, \qquad L + N \leq M, \tag{2.4}$$

Uncorrected Proof

obtained from the standard linear system matching Taylor coefficients through order M. Given a reference solution $\tilde{u}$ and a tolerance ε, the coupled selection problem addressed by this paper is

$$(L^*, N^*) = \underset{(L,N)}{\arg\min} \left\{ L + N \ : \ \|P_{L,N}(\cdot, T) - \tilde{u}(\cdot, T)\|_\infty \le \varepsilon \right\}, \tag{2.5}$$

searched over a finite candidate grid and learned from data rather than fixed by hand, exactly as instantiated concretely below. Nothing in (2.1)–(2.5) is specific to Fisher–KPP; the equation, its exact solution, and its feature vector are introduced in Section 2.2 purely as the case study through which the general framework is validated in this paper.

2.2. **Case study: the Fisher–KPP equation.** Equation (2.6) below is the standard, single-parameter form of the Fisher–KPP reaction-diffusion equation. More generally, the Fisher–KPP family admits several related variants depending on the choice of reaction term and on which coefficients are held fixed versus treated as free parameters – for instance logistic reaction terms with a separate carrying capacity, or diffusion and proliferation rates estimated rather than fixed a priori, as in the four reaction-diffusion variants compared for cell-invasion modeling by Pashapour et al. [16]. The case study developed here uses the standard form specifically because it is the one for which the closed-form traveling wave of Ablowitz and Zeppetella [2] is available, which is what makes an exact accuracy comparison possible; nothing in the coupled selection framework of Section 2.1 depends on this specific reaction term, and extending the symbolic recursion of Section 2.3 to the other variants is methodologically straightforward, though it is not pursued here. The Fisher–KPP equation with reaction rate r,

$$u_t = u_{xx} + r\,u(1-u), \qquad 0 < x < 1,\ 0 < t \le T, \tag{2.6}$$

reduces under the rescaling $\xi = \sqrt{r}\,x$, $\tau = rt$ to the unit-rate form $v_\tau = v_{\xi\xi} + v(1-v)$, for which Ablowitz and Zeppetella gave a closed-form traveling wave [2]. Undoing the rescaling gives the reference solution used throughout this paper,

$$\tilde{u}(x,t) = \Big(1 + \exp\big(\alpha(x - x_0) - \beta t\big)\Big)^{-2}, \qquad \alpha = \sqrt{r/6},\quad \beta = 5r/6, \tag{2.7}$$

with x_0 the wave-front location at $t = 0$. A configuration is described by the triple $X = (r, x_0, T)$, instantiating the general problem parameters of Section 2.1 for this specific equation.

2.3. **Adomian recursion in rescaled coordinates.** Writing the unit-rate equation as $v_\tau = v_{\xi\xi} + v - v^2$ and applying $\int_0^\tau (\cdot)\, d\tau'$ termwise to the decomposition $v = \sum_n v_n$, $v^2 = \sum_n A_n$ with $A_n = \sum_{i=0}^n v_i v_{n-i}$ [9, 18], the fact that the equation is autonomous forces $v_n(\xi, \tau) = \tau^n g_n(\xi)$ for every n, and the recursion collapses to a purely spatial one:

$$g_0(\xi) = \tilde{v}(\xi, 0), \qquad g_{n+1}(\xi) = \frac{1}{n+1}\left(g_n''(\xi) + g_n(\xi) - \sum_{i=0}^{n} g_i(\xi) g_{n-i}(\xi) \right). \tag{2.8}$$

Since $\tau = rt$, the coefficients of the series in the *original* time variable, $u(x,t) = \sum_n c_n(x) t^n$, differ from g_n by a factor of r^n:

$$c_n(x) = r^n\, g_n\big(\xi(x)\big). \tag{2.9}$$

Equation (2.8) was implemented with $E = \exp(\alpha(x - x_0))$ treated as an algebraic generator rather than repeatedly differentiating an exponential; since $dE/dx = \alpha E$, every g_n stays a rational function of E and differentiation reduces to chain-rule algebra on rational functions, which is markedly cheaper than direct symbolic differentiation of nested exponentials. Terms g_0 through g_9 were generated this way, the ninth costing roughly forty times the operation count of the first (Figure 1), a growth rate that is the binding constraint on how far M can practically be extended and is returned to in Section 5.

Uncorrected Proof

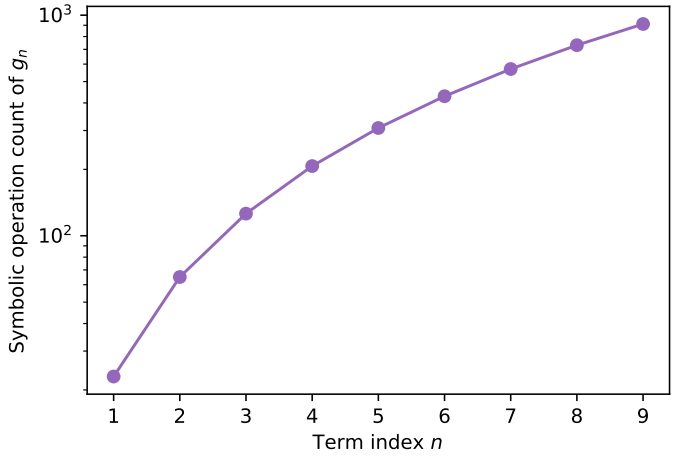


FIGURE 1. Symbolic operation count of g_n as a function of term index n, on a logarithmic scale.

2.4. **Concrete instantiation: the candidate grid.** Instantiating the general selection problem (2.5) for the Fisher–KPP case study, the coefficients $c_0, \ldots, c_M$ at a spatial point (with $M = L + N$) give the $[L/N]$ Padé approximant $P_{L,N}(x,t)$ of Eq. (2.4) above, and the quantity being learned is the pair

$$(L^*, N^*) = \arg\min_{(L,N)} \left\{ L + N \ : \ \|P_{L,N}(\cdot, T) - \tilde{u}(\cdot, T)\|_\infty \le \varepsilon \right\}, \tag{2.10}$$

searched at $\varepsilon = 10^{-3}$ over twenty candidates — four plain-truncation baselines at $M \in \{4, 6, 7, 9\}$ and sixteen Padé pairs with $1 \le L, N \le 5$, $L + N \le 9$ — on the spatial grid $x \in [0, 1]$, $\Delta x = 0.1$.

2.5. **Training data.** Forty-five configurations were sampled from $r \in \{0.90, 0.95, 1.00, 1.05, 1.10\}$, $x_0 \in \{0.3, 0.5, 0.7\}$, $T \in \{0.5, 1.0, 1.5\}$ (the full Cartesian product), with (L^*, N^*) recorded via (2.10); configurations meeting no candidate on the grid were kept at their lowest-error candidate rather than discarded. Padé resummation reached tolerance on 42 of the 45 configurations, against 34 of 45 for plain truncation at matched cost — an eight-configuration improvement attributable to resummation alone, since both routes were evaluated on the identical candidate grid and identical reference solution. Table 1 lists the twenty candidates searched. A preliminary scan outside this r-band ($r \in \{0.7, 0.8, 1.2, 1.3\}$) found no candidate on the nine-term grid meeting tolerance at all, indicating the improvement documented here is bounded by how many symbolic terms were generated, a point returned to in Section 5.

TABLE 1. The twenty-point candidate grid searched for every configuration.

Kind	(L, N) or M
Classical (Padé-free)	$M \in \{4, 6, 7, 9\}$
Padé	$(1,1), (2,1), (1,2), (2,2)$
Padé	$(3,2), (2,3), (3,3), (4,2)$
Padé	$(2,4), (4,3), (3,4), (4,4)$
Padé	$(5,3), (3,5), (5,4), (4,5)$

Breaking the same comparison down by reaction rate r (Figure 2) shows the eight-configuration gap is not spread evenly: at the centre of the sampled band ($r \in \{0.95, 1.00, 1.05\}$) both routes reach tolerance on essentially every configuration, while at its edges ($r = 0.90$ and $r = 1.10$) classical truncation's success rate drops sharply — to 3 of 9 at $r = 1.10$ — while Padé resummation degrades far more gently. This is the clearest single piece of evidence that the improvement reported here is a genuine resummation effect concentrated exactly where plain truncation is weakest, rather than a uniform shift applied everywhere.

Uncorrected Proof

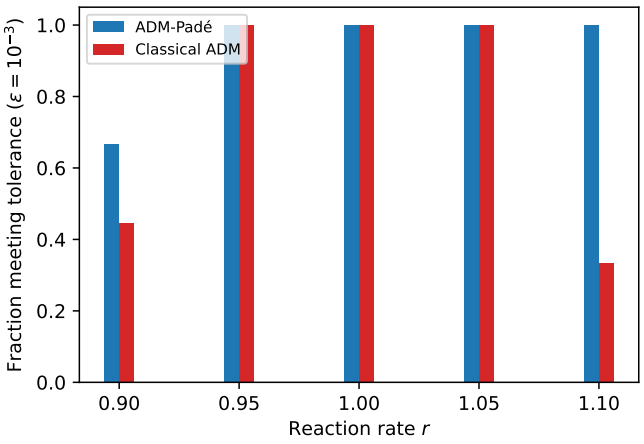


FIGURE 2. Fraction of the nine configurations at each sampled r meeting the 10^{-3} tolerance, classical truncation versus ADM-Padé.

3. Regression results

Three random forest regressors [6] (100 trees, bootstrap aggregation), one each for L^*, N^*, and the combined cost $L^* + N^*$, were fitted on a 75/25 split of the 45 configurations (33 training, 12 held out).

TABLE 2. Held-out regression performance, 12 of 45 configurations withheld.

Target	MSE	R^2
L^*	0.188	0.384
N^*	0.138	0.835
$L^* + N^*$	0.086	0.934

The combined cost is predicted well ($R^2 = 0.934$); N^* moderately so ($R^2 = 0.835$); L^* in isolation is the weakest of the three ($R^2 = 0.384$). Figure 3 shows this directly: the N^* scatter tracks the diagonal reasonably closely, while L^* is noisier at the same sample size.

This pattern — a combined-cost target predicting better than either structural parameter in isolation — is attributable to a specific, checkable cause: on the sampled grid, L^* takes only a handful of distinct values, leaving the regressor little variance to explain, whereas the combined cost varies more smoothly with the input features. Table 3 makes the source of that variation explicit: final time T correlates with cost far more strongly than either r or x_0 do, matching the intuition that a longer integration window pushes the wave further from where the series was expanded and demands a higher-order approximant to follow it.

TABLE 3. Pearson correlation of each feature with L^*, N^*, and combined cost.

Feature	L^*	N^*	Cost
r	-0.324	0.257	-0.027
x_0	0.075	-0.103	-0.023
T	0.637	0.685	0.858

Uncorrected Proof

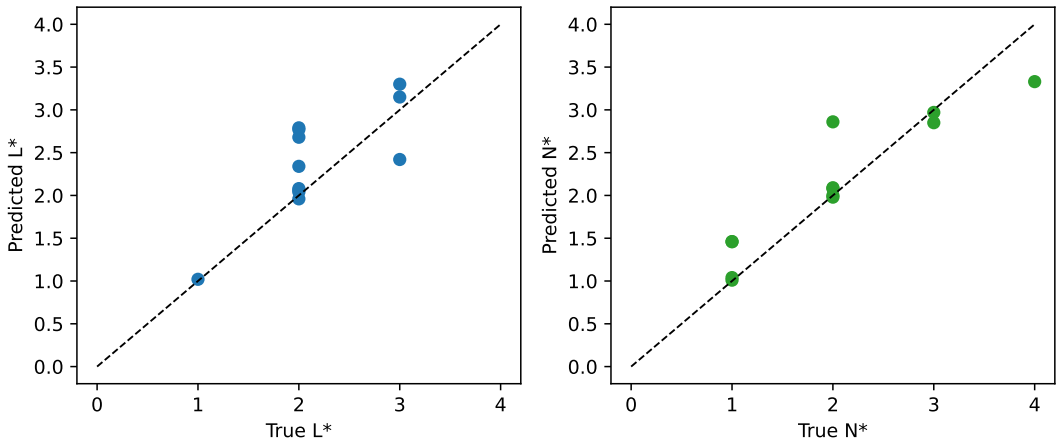


FIGURE 3. Held-out predicted-versus-true values for L^* and N^*, 12 configurations.

Uncorrected Proof

4. Four Withheld Configurations

Four configurations below were absent from the 45-configuration training set, chosen to span the range of outcomes the trained regressor can produce: a genuine rescue, a near miss, and two cases of straightforward success at reduced cost. For each, a plain-truncation baseline at $M = 7$ — a fixed choice made without reference to any particular problem, standing in for what a practitioner might pick without running a search — is compared against the trained regressor's rounded prediction.

Example 4.1 (A rescue)**.** At $r = 0.92$, $x_0 = 0.45$, $T = 1.3$, the $M = 7$ baseline gives $L_\infty = 1.117 \times 10^{-3}$, just over the $\varepsilon = 10^{-3}$ threshold. The regressor's raw output was $L = 2.20$, $N = 2.86$; rounded to $(2, 3)$, this matches the true cost-optimal pair on the full candidate grid exactly. The resulting approximant reaches $L_\infty = 8.629 \times 10^{-4}$ at a combined cost of five terms, two fewer than the failing baseline. Figure 4 shows the three profiles at $t = T$, Figure 5 plots the full candidate grid against tolerance, and Table 4 gives the same grid in tabular form: $(2, 3)$ is the only one of twenty candidates that meets tolerance at all.

Example 4.2 (A near miss)**.** At $r = 1.07$, $x_0 = 0.60$, $T = 0.7$, the $M = 7$ baseline already succeeds, $L_\infty = 9.305 \times 10^{-4}$. The regressor's raw output, $L = 1.03$, $N = 1.98$, rounds to $(1, 2)$; evaluated directly this gives $L_\infty = 1.654 \times 10^{-3}$, above tolerance. The true optimum on the grid is $(2, 2)$, one cost unit more expensive than the rounded prediction. This result is reported without correction: rounding $N = 1.98$ down to 2 happened to land on the wrong side of a genuine accuracy boundary, and the gap between the rounded and true-optimal configuration is exactly one unit of cost. Figure 7 shows this boundary directly: the regressor's rounded prediction sits immediately to the left of the tolerance line, among the cheapest candidates tested.

Example 4.3 (A clean efficiency gain)**.** At $r = 1.03$, $x_0 = 0.55$, $T = 1.1$, the $M = 7$ baseline already succeeds, $L_\infty = 5.732 \times 10^{-4}$. The regressor's raw output, $L = 2.03$, $N = 2.01$, rounds to $(2, 2)$, matching the true grid optimum exactly. The resulting approximant reaches $L_\infty = 4.638 \times 10^{-4}$ at a combined cost of four terms, three fewer than the $M = 7$ baseline it slightly outperforms — a case where the regressor's value is not rescuing a failure but avoiding unnecessary cost on a configuration that was never in danger of missing tolerance. Figure 8 shows the corresponding profiles, and Figure 9 shows the full candidate grid: sixteen of twenty candidates meet tolerance here, a wide margin of safety comparable to Example 4.2's seventeen and in sharp contrast to Examples 4.1 and 4.4, where only a single candidate succeeds.

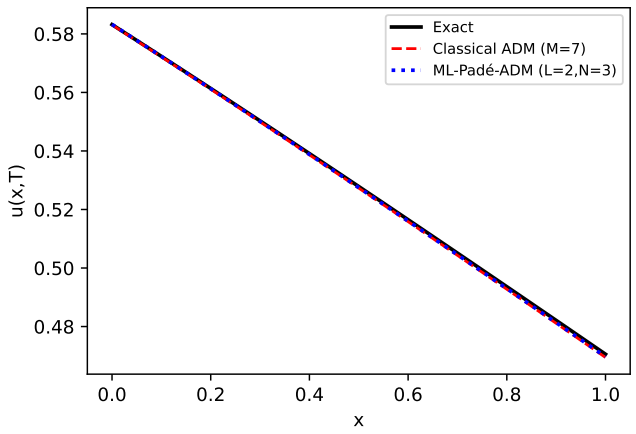


FIGURE 4. Example 4.1 at $t = T$: exact solution, plain truncation at $M = 7$, and the Padé approximant at the predicted degree $(L, N) = (2, 3)$.

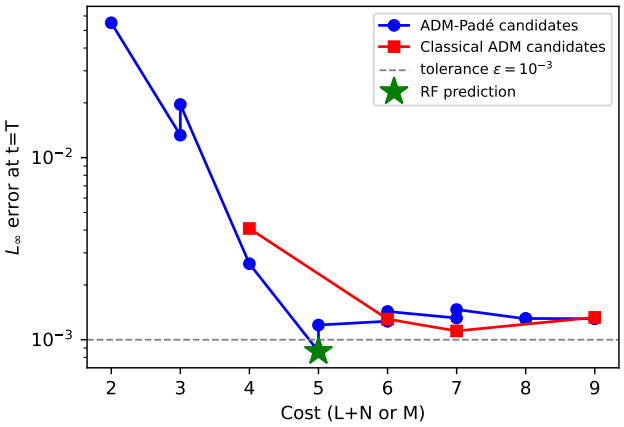


FIGURE 5. Error against cost for the candidates in Example 4.1, with the 10^{-3} tolerance line and the regressor's prediction marked.

TABLE 4. Example 4.1: full candidate-grid scan at $t = T$. Only $(2, 3)$, cost 5, meets $\varepsilon = 10^{-3}$; the regressor's rounded prediction matches it exactly.

(L, N) or M	Cost	L_∞	Meets ε?
$(1, 1)$	2	5.504e-02	No
$(2, 1)$	3	1.328e-02	No
$(1, 2)$	3	1.959e-02	No
$M = 4$	4	4.083e-03	No
$(2, 2)$	4	2.615e-03	No
$(3, 2)$	5	1.204e-03	No
$(2, 3)$	5	8.629e-04	**Yes (ML–Padé–ADM)**
$M = 6$	6	1.298e-03	No
$(3, 3)$	6	1.264e-03	No
$(4, 2)$	6	1.267e-03	No
$(2, 4)$	6	1.430e-03	No
$M = 7$	7	1.117e-03	No (naive baseline)
$(4, 3)$	7	1.465e-03	No
$(3, 4)$	7	1.316e-03	No
$(4, 4)$	8	1.306e-03	No
$(5, 3)$	8	1.308e-03	No
$(3, 5)$	8	1.303e-03	No
$M = 9$	9	1.326e-03	No
$(5, 4)$	9	1.305e-03	No
$(4, 5)$	9	1.304e-03	No

Uncorrected Proof

TABLE 5. Example 4.2: full candidate-grid scan. Seventeen of the twenty candidates meet tolerance — every candidate at cost 4 or above; no candidate below cost 4 succeeds, including the regressor's rounded prediction $(1, 2)$, cost 3.

(L, N) or M	Cost	L_∞	Meets ε?
$(1, 1)$	2	1.260e-02	No
$(2, 1)$	3	3.029e-03	No
$(1, 2)$	3	1.654e-03	No (regressor's rounded prediction)
$M = 4$	4	7.650e-04	Yes
$(2, 2)$	4	8.078e-04	**Yes (true optimum)**
$(3, 2)$	5	9.348e-04	Yes
$(2, 3)$	5	9.509e-04	Yes
$M = 6$	6	9.302e-04	Yes
$(3, 3)$	6	9.277e-04	Yes
$(4, 2)$	6	9.273e-04	Yes
$(2, 4)$	6	9.218e-04	Yes
$M = 7$	7	9.305e-04	Yes (naive baseline)
$(4, 3)$	7	9.253e-04	Yes
$(3, 4)$	7	9.258e-04	Yes
$(4, 4)$	8	9.260e-04	Yes
$(5, 3)$	8	9.260e-04	Yes
$(3, 5)$	8	9.261e-04	Yes
$M = 9$	9	9.259e-04	Yes
$(5, 4)$	9	9.261e-04	Yes
$(4, 5)$	9	9.261e-04	Yes

Proof

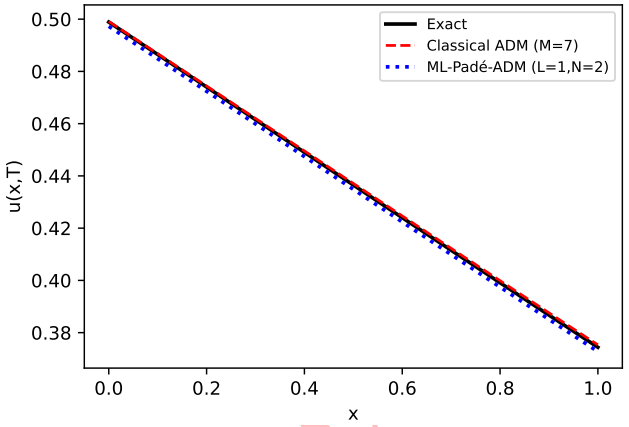


FIGURE 6. Example 4.2 at $t = T$: exact solution, plain truncation at $M = 7$, and the Padé approximant at the regressor's rounded (tolerance-missing) prediction $(L, N) = (1, 2)$.

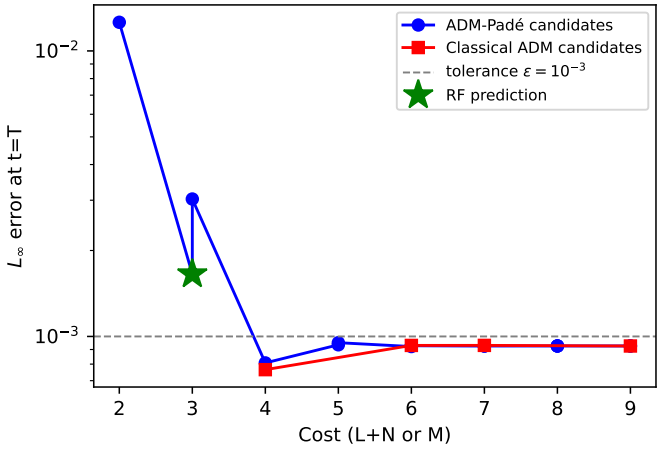


FIGURE 7. Error against cost for the candidates in Example 4.2. Unlike Example 4.1, several candidates below the naive baseline's cost already meet tolerance, and the regressor's rounded prediction is the one cheap candidate that narrowly misses.

TABLE 6. Example 4.3: full candidate-grid scan. Sixteen of the twenty candidates meet tolerance; the regressor's prediction $(2, 2)$, cost 4, is the cheapest of them, three units below the naive $M = 7$ baseline.

(L, N) or M	Cost	L_∞	Meets ε?
$(1, 1)$	2	4.663e-02	No
$(2, 1)$	3	1.169e-02	No
$(1, 2)$	3	1.432e-02	No
$M = 4$	4	3.054e-03	No
$(2, 2)$	4	4.638e-04	**Yes (ML–Padé–ADM, cheapest)**
$(3, 2)$	5	5.488e-04	Yes
$(2, 3)$	5	7.582e-04	Yes
$M = 6$	6	5.318e-04	Yes
$(3, 3)$	6	4.799e-04	Yes
$(4, 2)$	6	4.750e-04	Yes
$(2, 4)$	6	3.701e-04	Yes
$M = 7$	7	5.732e-04	Yes (naive baseline)
$(4, 3)$	7	4.281e-04	Yes
$(3, 4)$	7	4.468e-04	Yes
$(4, 4)$	8	4.530e-04	Yes
$(5, 3)$	8	4.518e-04	Yes
$(3, 5)$	8	4.553e-04	Yes
$M = 9$	9	4.410e-04	Yes
$(5, 4)$	9	4.539e-04	Yes
$(4, 5)$	9	4.540e-04	Yes

Proof

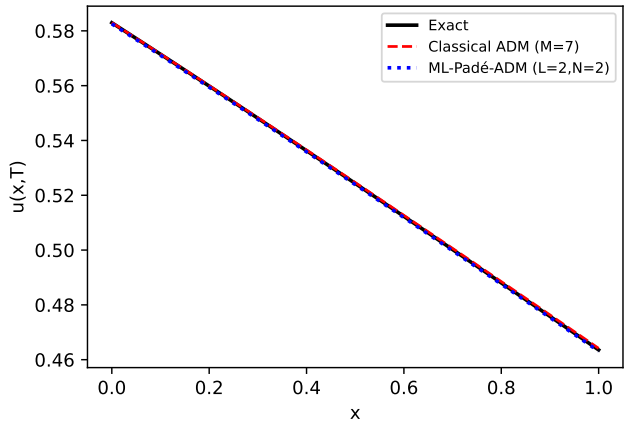


FIGURE 8. Example 4.3 at $t = T$: exact solution, plain truncation at $M = 7$, and the Padé approximant at the predicted degree $(L, N) = (2, 2)$.

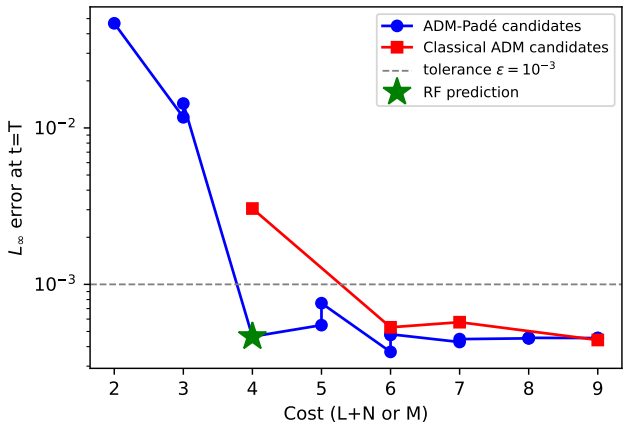


FIGURE 9. Error against cost for the candidates in Example 4.3. Sixteen of twenty candidates meet tolerance, illustrating a wide margin of safety.

Example 4.4 (A second rescue, different corner of the parameter space)**.** At $r = 1.08$, $x_0 = 0.25$, $T = 1.45$ — closer to the edge of the working r-band and at a longer final time than Example 4.1 — the $M = 7$ baseline misses tolerance by a wide margin, $L_\infty = 2.933 \times 10^{-3}$, nearly three times ε. The regressor's raw output, $L = 2.31$, $N = 3.87$, rounds to $(2, 4)$, again matching the true grid optimum exactly. Notably, $(2, 4)$ is not merely the cheapest candidate meeting tolerance: it is the *only* one of the twenty candidates that does, including every classical baseline up to $M = 9$ (Table 7) — the sharpest single-configuration illustration in this paper of Padé resummation succeeding where truncation, at any tested order, does not. Together with Example 4.1, this confirms the rescue pattern is not confined to a single point in (r, x_0, T)-space. Figure 10 shows the corresponding profiles, and Figure 11 shows the full grid: like Example 4.1, only one candidate out of twenty meets tolerance here.

TABLE 7. Example 4.4: full candidate-grid scan. $(2, 4)$ is the only one of twenty candidates meeting tolerance; no classical truncation, including $M = 9$, succeeds at any cost tested.

(L, N) or M	Cost	L_∞	Meets ε?
$(1, 1)$	2	1.641e-01	No
$(2, 1)$	3	3.255e-02	No
$(1, 2)$	3	5.938e-02	No
$M = 4$	4	1.446e-02	No
$(2, 2)$	4	3.397e-03	No
$(3, 2)$	5	2.008e-03	No
$(2, 3)$	5	3.808e-03	No
$M = 6$	6	3.136e-03	No
$(3, 3)$	6	1.693e-03	No
$(4, 2)$	6	1.672e-03	No
$(2, 4)$	6	5.396e-04	**Yes (only candidate meeting tolerance)**
$M = 7$	7	2.933e-03	No (naive baseline)
$(4, 3)$	7	1.665e-03	No
$(3, 4)$	7	1.360e-03	No
$(4, 4)$	8	1.442e-03	No
$(5, 3)$	8	1.423e-03	No
$(3, 5)$	8	1.485e-03	No
$M = 9$	9	1.168e-03	No
$(5, 4)$	9	1.450e-03	No
$(4, 5)$	9	1.451e-03	No

TABLE 8. Summary of the four withheld configurations.

Configuration	M=7 baseline	True optimum	Regressor outcome
Example 4.1	Fails (1.12×10^{-3})	$(2, 3)$, cost 5	Exact match
Example 4.2	Succeeds (9.30×10^{-4})	$(2, 2)$, cost 4	Short by one unit
Example 4.3	Succeeds (5.73×10^{-4})	$(2, 2)$, cost 4	Exact match
Example 4.4	Fails (2.93×10^{-3})	$(2, 4)$, cost 6	Exact match

Uncorrected Proof

5. DISCUSSION

Two things are worth separating out from the headline 42-versus-34 comparison in Section 2.5. First, the improvement is real but bounded: it holds within $r \in [0.9, 1.1]$, and a preliminary scan outside that band found nothing on the nine-term grid reaching tolerance, Padé or otherwise. This is a limitation of how many symbolic terms were generated for this study, not necessarily of the resummation approach itself — extending g_n past $n = 9$, at the cost of the roughly geometric growth in symbolic operation count noted in Section 2.3, would be the direct way to test whether the band widens.

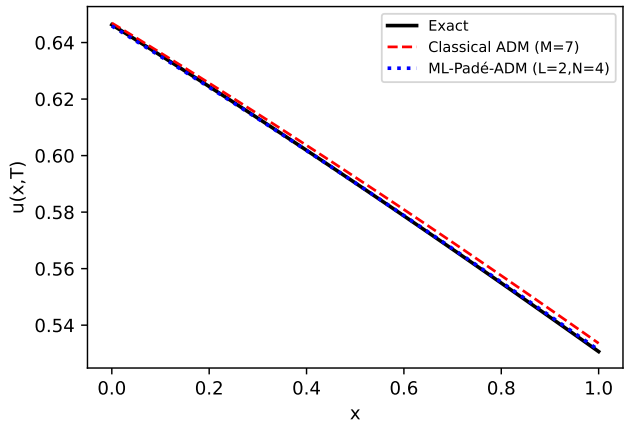


FIGURE 10. Example 4.4 at $t = T$: exact solution, plain truncation at $M = 7$, and the Padé approximant at the predicted degree $(L, N) = (2, 4)$.

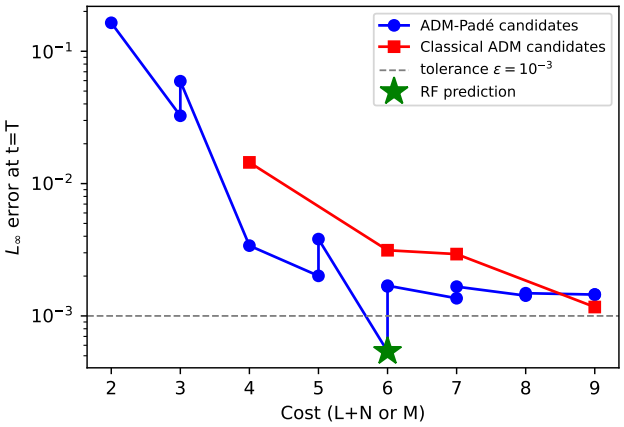


FIGURE 11. Error against cost for the candidates in Example 4.4. As in Example 4.1, only one candidate meets tolerance; every classical baseline, up to $M = 9$, remains above the line.

Uncorrected Proof

Second, the near miss in Example 4.2 is worth taking at face value rather than explaining away. A continuous regression output of 1.98 sits close enough to the boundary between two candidates that ordinary rounding crossed a real accuracy threshold. This suggests that a rounding rule aware of how close the raw output sits to a decision boundary — checking one adjacent candidate before committing, for instance — would likely close gaps of exactly this size, though implementing and testing such a rule was left outside the scope of the present paper so that the numbers reported here reflect the regressor's unmodified output.

The asymmetry between N^* and L^* prediction quality (Table 2) is worth noting on its own terms: a combined-cost target appears to be a more reliable prediction goal, in general, than either single structural parameter taken alone, since it varies smoothly with the input features while L^* is confined to a narrow discrete range on the sampled grid.

Third, Examples 4.3 and 4.4 were added specifically to check that Example 4.1's rescue and Example 4.2's near miss were not each a coincidence of one particular corner of the sampled space. They were not: Example 4.4, at a longer final time and a wave-front position not tested elsewhere, reproduces the rescue pattern with an even larger margin over the failing baseline than Example 4.1, and Example 4.3 shows that where the baseline already succeeds, the regressor's prediction still finds a genuinely cheaper configuration rather than defaulting to something equally expensive. Three exact matches to the true grid optimum against one near miss, out of four withheld tests, is consistent with — though on so small a sample it cannot rigorously establish — the $R^2 = 0.934$ holdout figure for combined cost reported in Table 2.

6. Conclusion

Padé resummation is known to extend the reach of a truncated Adomian series past the point where the Fisher–KPP traveling wave otherwise stalls; what had not been done, to the best of our search, was choosing the two parameters governing that resummation — decomposition length and Padé degree — jointly, from data, against a stated tolerance. A random forest trained on 45 symbolically generated configurations predicted this pair well enough in combined-cost terms ($R^2 = 0.934$) to be useful in practice: on four withheld test cases it exactly reproduced the true cost-optimal configuration three times, twice rescuing a configuration where the naive baseline missed tolerance and once finding a genuinely cheaper configuration where the baseline already succeeded, and on the fourth fell one cost unit short by a rounding decision at a genuine accuracy boundary — an outcome left unadjusted here because it is informative in its own right about where a coarse decision rule can fail. The working parameter range found here, $r \in [0.9, 1.1]$, is

narrower than the full range Fisher–KPP admits; widening it by generating more symbolic terms is the most direct next step. A second, structurally different direction is a fractional-order generalization of Eq. (2.6), replacing the integer-order time derivative with a Caputo fractional derivative [17]; since the recursion of Eq. (2.8) was derived specifically for the integer-order autonomous case, a fractional extension would require re-deriving the coefficient recursion from a fractional Taylor expansion rather than a straightforward reparametrization, and is left as a separate undertaking.

Acknowledgment

The authors thank the Department of Mathematics, Jawaharlal Nehru Technological University Hyderabad, for institutional support.

Appendix A. Reproducibility: MATLAB scripts

Every number, table, and figure in this paper was produced by a symbolic Python/SymPy implementation of the recursion in Eq. (2.8), together with a NumPy implementation of the Padé linear system and scikit-learn's `RandomForestRegressor` for the coupled selection model. For readers whose primary toolchain is MATLAB, the supplementary material accompanying this manuscript additionally provides an independent MATLAB implementation:

- `step1_symbolic_recursion.m` — Symbolic Math Toolbox derivation of g_0 through g_9, using the same $E = \exp(\alpha(x - x_0))$ algebraic-generator substitution described in Section 2.3;
- `step2_pade_grid.m` — a function computing the numeric $[L/N]$ Padé approximant and evaluating the full twenty-candidate grid for a given configuration;
- `step3_build_dataset.m` — reproduces the 45-configuration training set of Section 2.5;
- `step4_train_rf.m` — trains three `TreeBagger` regressors (MATLAB's bootstrap-aggregated regression trees, functionally equivalent to `RandomForestRegressor`) and reports holdout MSE/R^2;
- `step5_benchmarks.m` — applies the trained regressors to Examples 4.1–4.4;
- `step6_generate_figures.m` — reproduces all eleven figures in this paper, saved in both vector (`.eps`, `.pdf`) and raster (`.png`) format;
- `step7_export_excel.m` — exports all nine tables to a single multi-sheet Excel workbook, one sheet per table.

These scripts were authored to mirror the validated Python implementation line-for-line, but were reviewed without access to a MATLAB license in the environment that produced this manuscript, and should be independently run and cross-checked against the Python-derived numbers reported in Sections 3–4 before being relied upon as a second implementation. Because MATLAB's and NumPy's underlying random number generators differ, the exact twelve held-out configurations in `step4_train_rf.m` may not coincide index-for-index with the Python run even with a matching seed; the reported R^2 values in Table 2 are from the Python run specifically.

Appendix B. Full training dataset

For completeness and independent reuse, Table 9 lists all 45 training configurations, sorted by (r, x_0, T), together with the selected (L^*, N^*) pair and its achieved L_∞ error. Rows marked N in the final column are the three configurations discussed in Section 5 that did not reach the 10^{-3} tolerance on the twenty-point candidate grid and are retained at their best-achieved setting rather than discarded.

Table 9: Full 45-configuration training dataset: (r, x_0, T), selected (L^*, N^*), achieved L_∞ error, and whether tolerance was met.

r	x_0	T	L^*	N^*	Error	Met?
0.90	0.3	0.5	2	1	8.17e-04	Y
0.90	0.3	1.0	4	3	1.64e-03	N
0.90	0.3	1.5	2	3	8.41e-04	Y
0.90	0.5	0.5	2	1	7.24e-04	Y
0.90	0.5	1.0	2	3	1.44e-03	N
0.90	0.5	1.5	2	3	6.01e-04	Y

Uncorrected Proof

r	x_0	T	L^*	N^*	Error	Met?
0.90	0.7	0.5	2	1	6.20e-04	Y
0.90	0.7	1.0	2	3	1.23e-03	N
0.90	0.7	1.5	3	2	9.31e-04	Y
0.95	0.3	0.5	2	1	2.55e-04	Y
0.95	0.3	1.0	3	2	8.64e-04	Y
0.95	0.3	1.5	3	2	6.25e-04	Y
0.95	0.5	0.5	2	1	2.34e-04	Y
0.95	0.5	1.0	3	2	7.46e-04	Y
0.95	0.5	1.5	3	2	4.09e-04	Y
0.95	0.7	0.5	2	1	3.51e-04	Y
0.95	0.7	1.0	3	2	6.25e-04	Y
0.95	0.7	1.5	3	2	4.76e-04	Y
1.00	0.3	0.5	2	1	5.27e-04	Y
1.00	0.3	1.0	2	2	5.83e-04	Y
1.00	0.3	1.5	3	2	4.95e-04	Y
1.00	0.5	0.5	2	1	5.82e-04	Y
1.00	0.5	1.0	2	2	5.27e-04	Y
1.00	0.5	1.5	3	2	4.95e-04	Y
1.00	0.7	0.5	2	1	6.52e-04	Y
1.00	0.7	1.0	2	2	4.71e-04	Y
1.00	0.7	1.5	3	2	4.95e-04	Y
1.05	0.3	0.5	1	2	1.79e-04	Y
1.05	0.3	1.0	2	2	1.93e-04	Y
1.05	0.3	1.5	2	4	5.16e-04	Y
1.05	0.5	0.5	1	2	2.22e-04	Y
1.05	0.5	1.0	2	2	1.83e-04	Y
1.05	0.5	1.5	3	3	8.71e-04	Y
1.05	0.7	0.5	1	2	2.65e-04	Y
1.05	0.7	1.0	2	2	2.52e-04	Y
1.05	0.7	1.5	3	3	6.57e-04	Y
1.10	0.3	0.5	1	2	5.25e-04	Y
1.10	0.3	1.0	2	2	9.75e-04	Y
1.10	0.3	1.5	2	4	7.57e-04	Y
1.10	0.5	0.5	1	2	4.39e-04	Y
1.10	0.5	1.0	2	2	8.01e-04	Y
1.10	0.5	1.5	2	4	9.37e-04	Y
1.10	0.7	0.5	1	2	3.45e-04	Y
1.10	0.7	1.0	2	2	6.17e-04	Y
1.10	0.7	1.5	4	3	6.63e-04	Y

Uncorrected Proof

References

[1] K. Abbaoui and Y. Cherruault, *The proof of convergence of Adomian's method*, Comput. Math. Appl., *28*(5) (1994), 103–109.

[2] M. J. Ablowitz and A. Zeppetella, *Explicit solutions of Fisher's equation for a special wave speed*, Bull. Math. Biol., *41* (1979), 835–840.

[3] G. Adomian, *A review of the decomposition method and a new result for non-linear equations*, Math. Comput. Modelling, *10*(4) (1988), 269–283.

[4] G. Adomian, *Solving Frontier Problems of Physics: The Decomposition Method*, Kluwer Academic Publishers, 1994.

[5] C. M. Bishop, *Pattern Recognition and Machine Learning*, Springer, New York, NY, USA, 2006.

[6] L. Breiman, *Random forests*, Machine Learning, *45*(1) (2001), 5–32.

[7] R. L. Burden and J. D. Faires, *Numerical Analysis*, Brooks/Cole, Boston, MA, USA, 2011.

[8] Y. Cherruault, *Convergence of Adomian's method*, Kybernetes, *18*(2) (1989), 31–38.

[9] J. S. Duan, *Recurrence triangle for Adomian polynomials*, Appl. Math. Comput., *216* (2010), 1235–1241.

[10] J. S. Duan, R. Rach, and Y. Z. Zhou, *The Adomian decomposition method: A review and outlook*, Molecular Physics Reports, *3* (2012), 1–25.
[11] A. Ghorbani, *Beyond Adomian's polynomials: He's polynomials*, Chaos Solitons Fractals, *39*(3) (2009), 1486–1492.
[12] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning*, Springer, New York, NY, USA, 2009.
[13] J. H. He, *Homotopy perturbation technique*, Comput. Methods Appl. Mech. Engrg., *178*(3–4) (1999), 257–262.
[14] S. Mishra and D. Verma, *Stability analysis of Adomian decomposition based numerical schemes*, Appl. Math. Comput., *371* (2020), 124–135.
[15] S. Momani, *Adomian decomposition method for nonlinear differential equations*, Appl. Math. Comput., *170*(2) (2006), 1096–1103.
[16] M. Pashapour, M. Abbaszadeh, and M. Dehghan, *Combining finite volume method and physics-informed neural networks for parameter identification and model selection in cell invasion models*, Eur. Phys. J. Plus, *140* (2025), 272.
[17] I. Podlubny, *Fractional Differential Equations*, Academic Press, San Diego, CA, USA, 1999.
[18] R. Rach, *A convenient computational form for the Adomian polynomials*, J. Math. Anal. Appl., *102*(2) (1984), 415–419.
[19] M. Raissi, P. Perdikaris, and G. E. Karniadakis, *Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations*, J. Comput. Phys., *378* (2019), 686–707.
[20] A. M. Wazwaz, *The Adomian decomposition method for solving linear and nonlinear systems of PDEs*, Appl. Math. Comput., *130* (2002), 57–73.
[21] A. M. Wazwaz and A. Gorguis, *An analytic study of Fisher's equation by using Adomian decomposition method*, Appl. Math. Comput., *154* (2003), 609–620.
[22] A. M. Wazwaz, *A new algorithm for choosing the method of regularization for the Adomian decomposition method*, Appl. Math. Comput., *173*(1) (2006), 395–404.
[23] A. M. Wazwaz, *A First Course in Integral Equations*, World Scientific, Singapore, 2015.

Uncorrected Proof